**Name: Maaz Sajid**

**Roll no:B23F0001CS061**

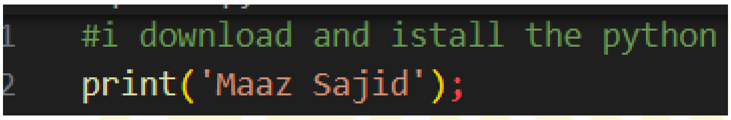
**Intern ID: : TN/IN02/PY/036**

**Task no: week 1 task**

**Question: 1. Install Python & print version.**

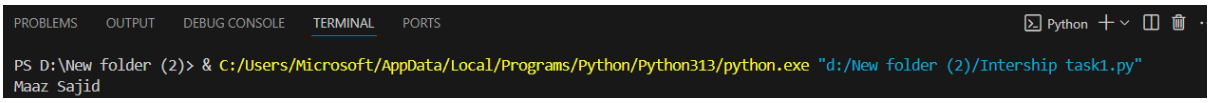
**I install successfully**

**2. Run hello script printing your name.**



```
#i download and istall the python
print('Maaz Sajid');
```

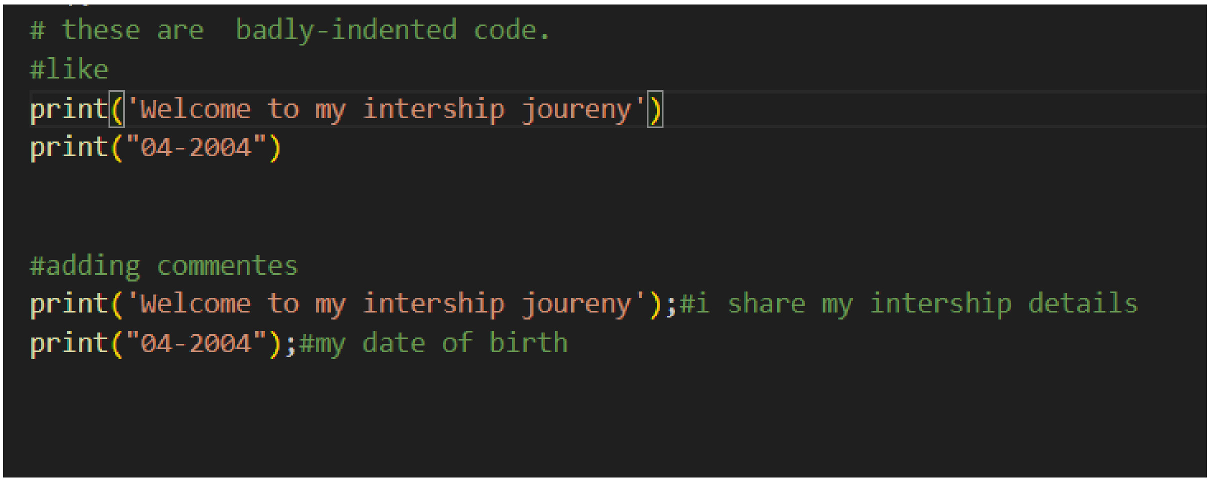
**Output:**



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/Intership task1.py"
Maaz Sajid
```

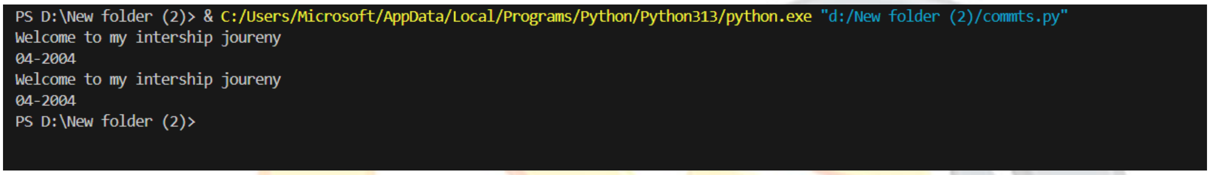
## Task #02:Week 1 Tasks – Syntax & Indentation

**Q#01:Fix badly-indented code.**



```
# these are  badly-indented code.
#like
print('Welcome to my intership joureny')
print("04-2004")


#adding commentes
print('Welcome to my intership joureny');#i share my intership details
print("04-2004");#my date of birth
```

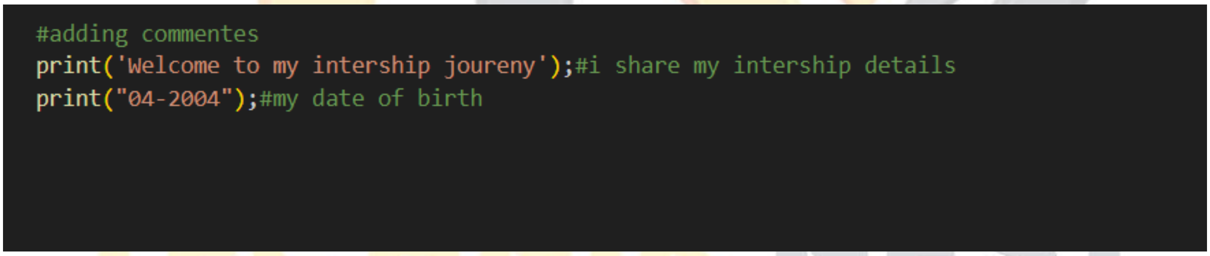
## Solution:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/commts.py"
Welcome to my intership joureny
04-2004
Welcome to my intership joureny
04-2004
PS D:\New folder (2)>
```

## 2. Add comments explaining each step.

## Solution:

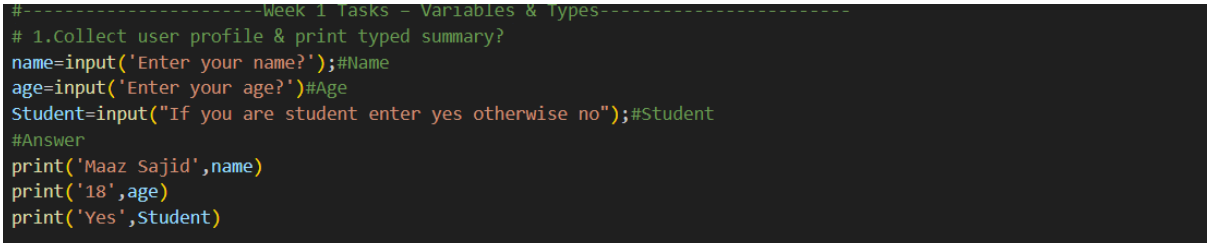


```
#adding commentes
print('Welcome to my intership joureny');#i share my intership details
print("04-2004");#my date of birth
```

Task#03 :Week 1 Tasks – Variables & Types

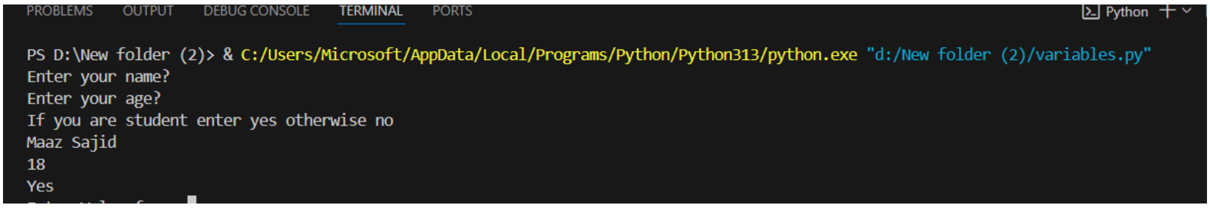
Q#1:Collect user profile & print typed summary?

Code:

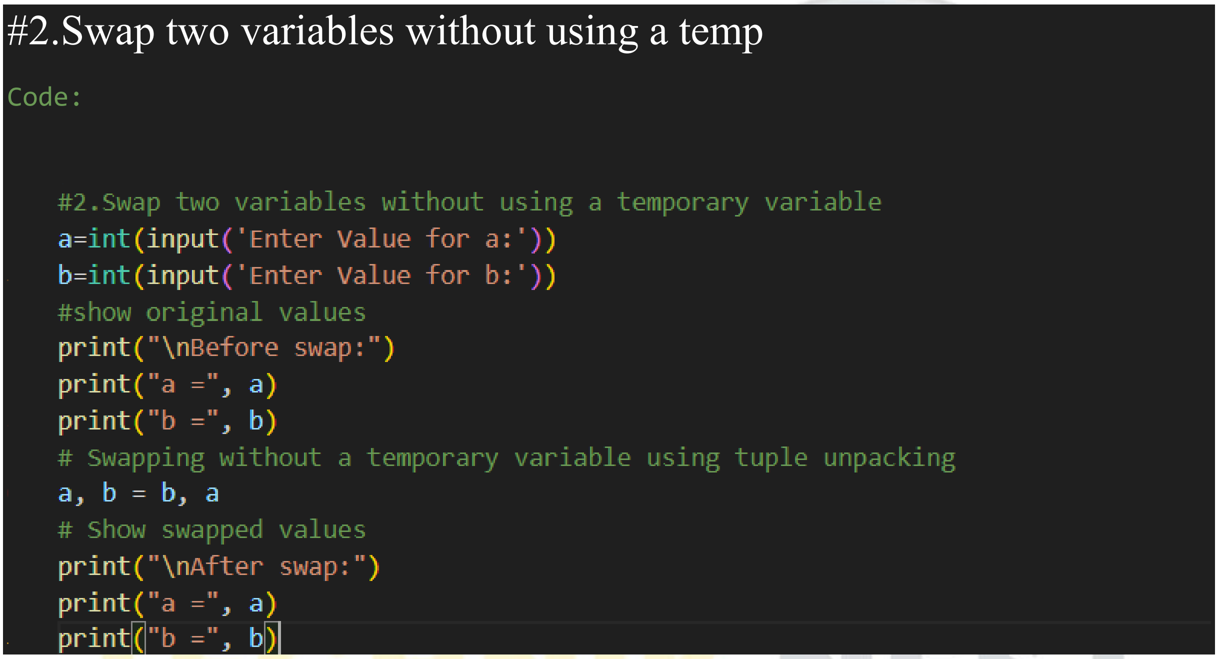


```
#-----------------------Week 1 Tasks - Variables & Types-----------------------
# 1.Collect user profile & print typed summary?
name=input('Enter your name?');#Name
age=input('Enter your age?')#Age
Student=input("If you are student enter yes otherwise no");#Student
#Answer
print('Maaz Sajid',name)
print('18',age)
print('Yes',Student)
```

Output:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/variables.py"
Enter your name?
Enter your age?
If you are student enter yes otherwise no
Maaz Sajid
18
Yes
```

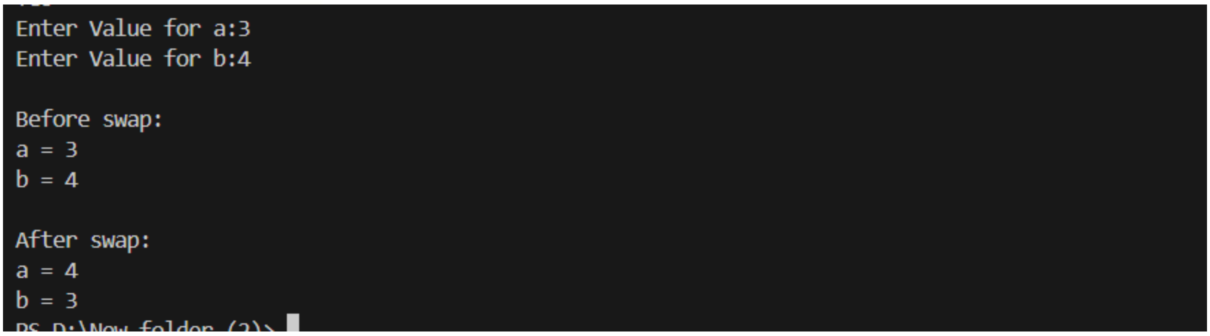


#2.Swap two variables without using a temp

Code:

```
#2.Swap two variables without using a temporary variable
a=int(input('Enter Value for a:'))
b=int(input('Enter Value for b:'))
#show original values
print("\nBefore swap:")
print("a =", a)
print("b =", b)
# Swapping without a temporary variable using tuple unpacking
a, b = b, a
# Show swapped values
print("\nAfter swap:")
print("a =", a)
print("b =", b)
```

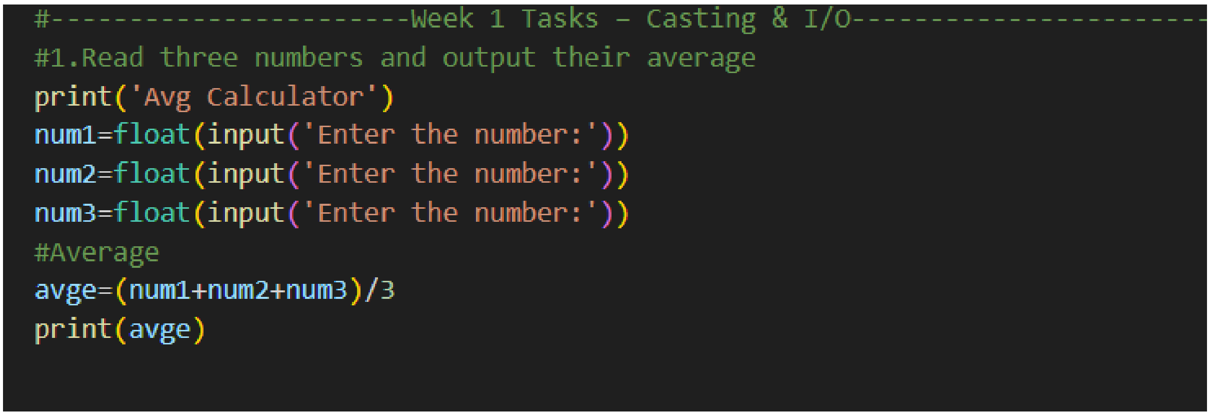
Solution



```
Enter Value for a:3
Enter Value for b:4

Before swap:
a = 3
b = 4

After swap:
a = 4
b = 3
PS D:\New folder (2)>
```

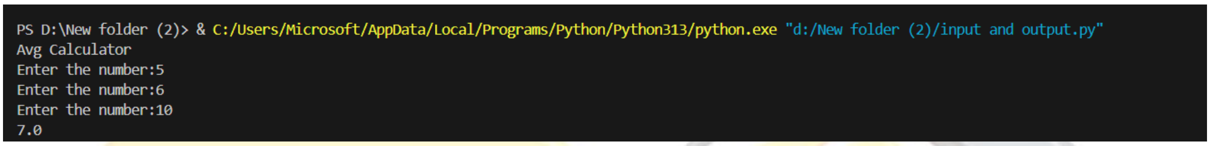
Task#04: **Week 1 Tasks – Casting & I/O**

**Q#1: Read three numbers; output avg.**



```
#-----------------------Week 1 Tasks – Casting & I/O-----------------------
#1.Read three numbers and output their average
print('Avg Calculator')
num1=float(input('Enter the number:'))
num2=float(input('Enter the number:'))
num3=float(input('Enter the number:'))
#Average
avge=(num1+num2+num3)/3
print(avge)
```

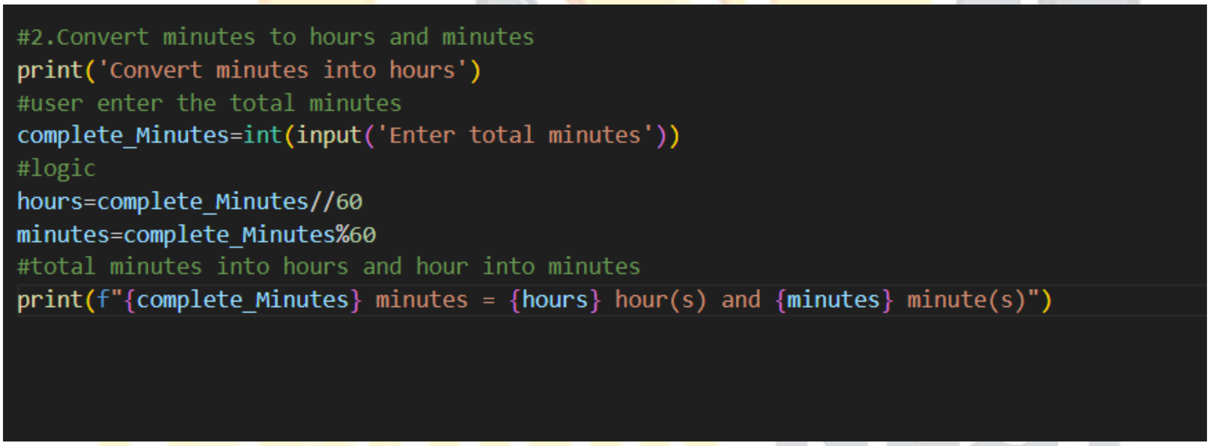
## Solution:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/input and output.py"
Avg Calculator
Enter the number:5
Enter the number:6
Enter the number:10
7.0
```

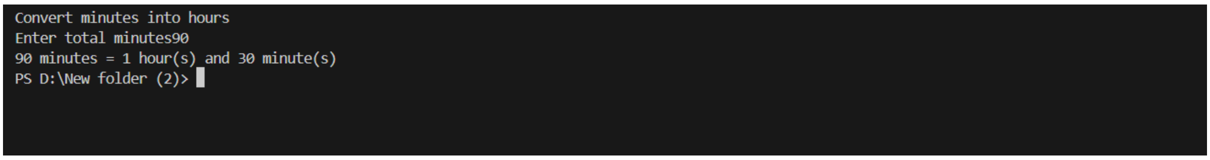
## 2. Convert minutes to hours + minutes.

## Code:



```
#2.Convert minutes to hours and minutes
print('Convert minutes into hours')
#user enter the total minutes
complete_Minutes=int(input('Enter total minutes'))
#logic
hours=complete_Minutes//60
minutes=complete_Minutes%60
#total minutes into hours and hour into minutes
print(f"{complete_Minutes} minutes = {hours} hour(s) and {minutes} minute(s)")
```

## Solution:

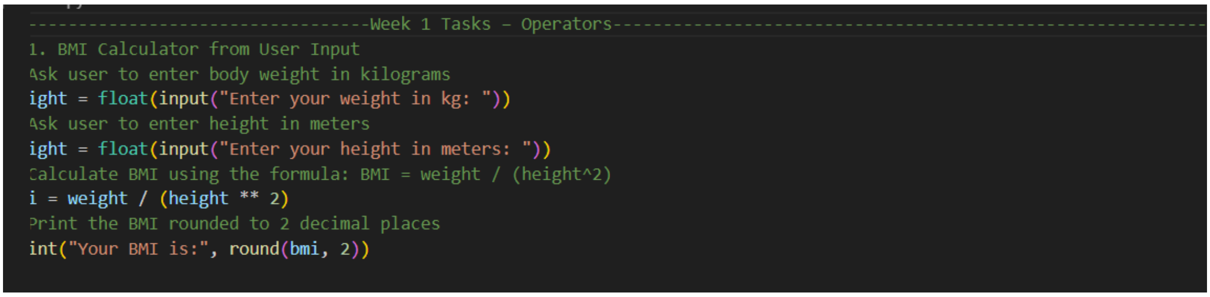


```
Convert minutes into hours
Enter total minutes90
90 minutes = 1 hour(s) and 30 minute(s)
PS D:\New folder (2)> 
```

## Task#05:Week 1 Tasks – Operators

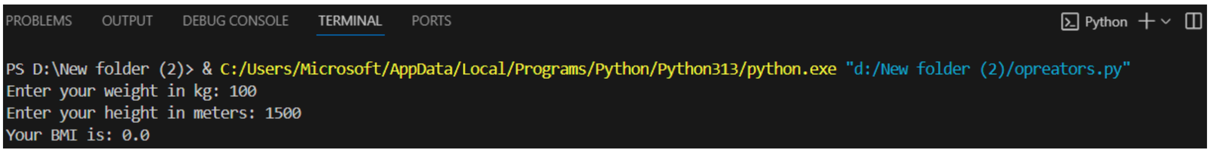
## Q#01: BMI calc from user input.

## Code:



```
--------------------------------Week 1 Tasks - Operators--------------------------------------------------------------
1. BMI Calculator from User Input
Ask user to enter body weight in kilograms
ight = float(input("Enter your weight in kg: "))
Ask user to enter height in meters
ight = float(input("Enter your height in meters: "))
Calculate BMI using the formula: BMI = weight / (height^2)
i = weight / (height ** 2)
Print the BMI rounded to 2 decimal places
int("Your BMI is:", round(bmi, 2))
```

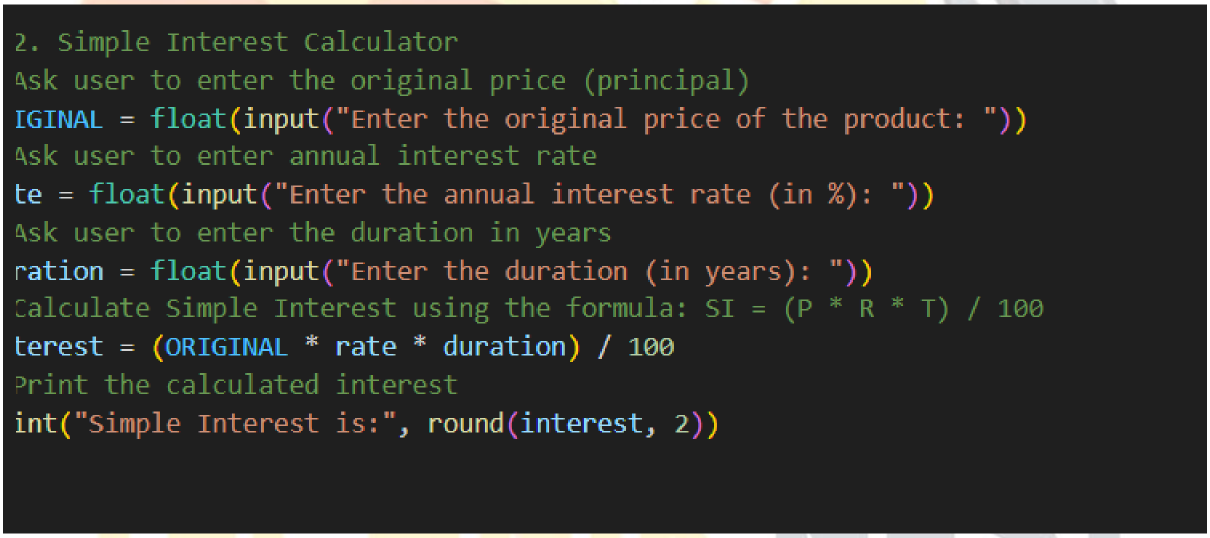
## Solution:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/opreators.py"
Enter your weight in kg: 100
Enter your height in meters: 1500
Your BMI is: 0.0
```

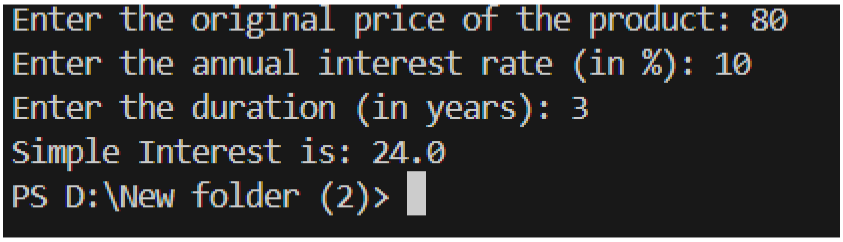
## 2. Simple interest calc.

## Code:



```
2. Simple Interest Calculator
Ask user to enter the original price (principal)
IGINAL = float(input("Enter the original price of the product: "))
Ask user to enter annual interest rate
te = float(input("Enter the annual interest rate (in %): "))
Ask user to enter the duration in years
ration = float(input("Enter the duration (in years): "))
Calculate Simple Interest using the formula: SI = (P * R * T) / 100
terest = (ORIGINAL * rate * duration) / 100
Print the calculated interest
int("Simple Interest is:", round(interest, 2))
```

## Solution:

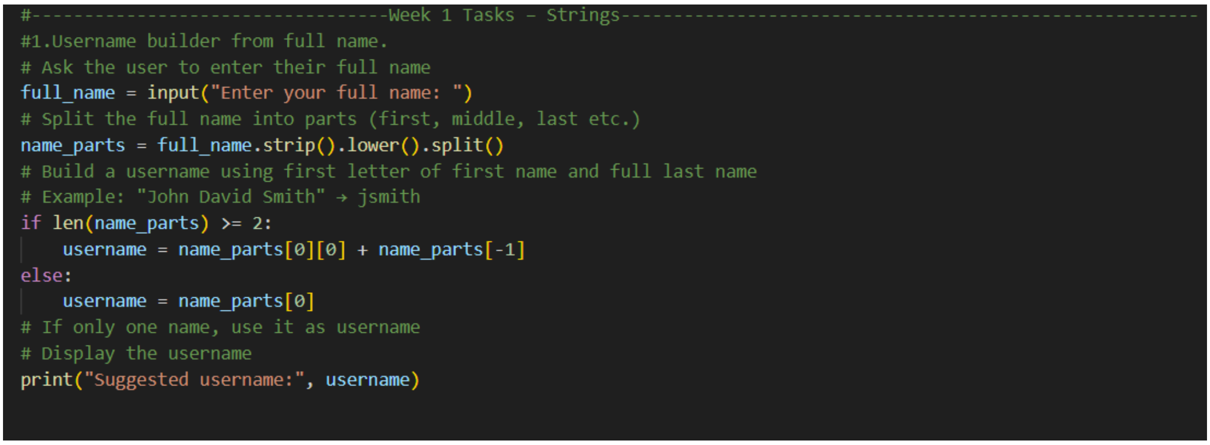


```
Enter the original price of the product: 80
Enter the annual interest rate (in %): 10
Enter the duration (in years): 3
Simple Interest is: 24.0
PS D:\New folder (2)> 
```

## Task#06:Week 1 Tasks – Strings

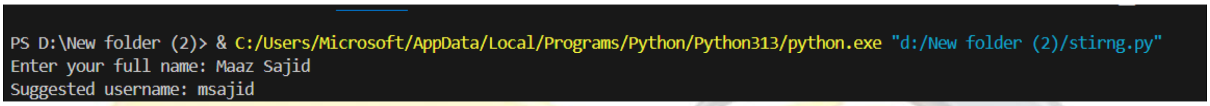
## Q#01:Username builder from full name.

## Code:



```
#---------------------------------Week 1 Tasks – Strings-------------------------------------------------------
#1.Username builder from full name.
# Ask the user to enter their full name
full_name = input("Enter your full name: ")
# Split the full name into parts (first, middle, last etc.)
name_parts = full_name.strip().lower().split()
# Build a username using first letter of first name and full last name
# Example: "John David Smith" → jsmith
if len(name_parts) >= 2:
    username = name_parts[0][0] + name_parts[-1]
else:
    username = name_parts[0]
# If only one name, use it as username
# Display the username
print("Suggested username:", username)
```

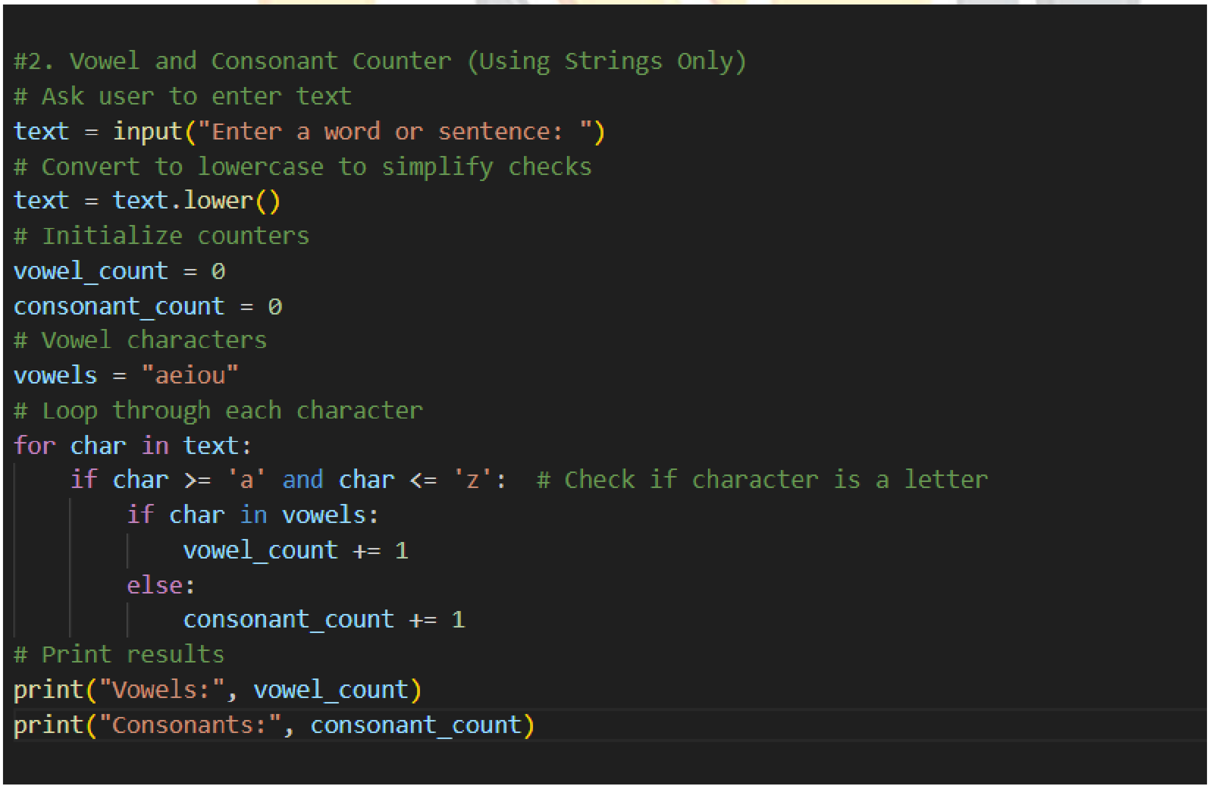
## Solution:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/stirng.py"
Enter your full name: Maaz Sajid
Suggested username: msajid
```

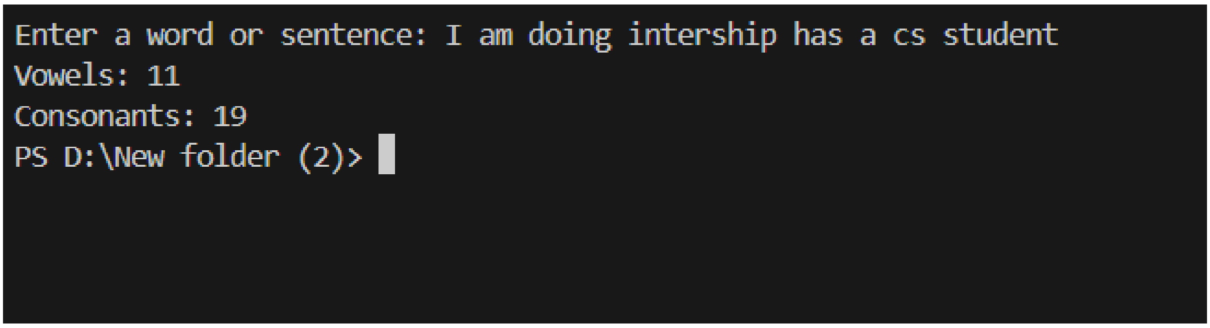
## Q#02: Vowel/consonant counter.

## Code:



```
#2. Vowel and Consonant Counter (Using Strings Only)
# Ask user to enter text
text = input("Enter a word or sentence: ")
# Convert to lowercase to simplify checks
text = text.lower()
# Initialize counters
vowel_count = 0
consonant_count = 0
# Vowel characters
vowels = "aeiou"
# Loop through each character
for char in text:
    if char >= 'a' and char <= 'z':  # Check if character is a letter
        if char in vowels:
            vowel_count += 1
        else:
            consonant_count += 1
# Print results
print("Vowels:", vowel_count)
print("Consonants:", consonant_count)
```

## Output:

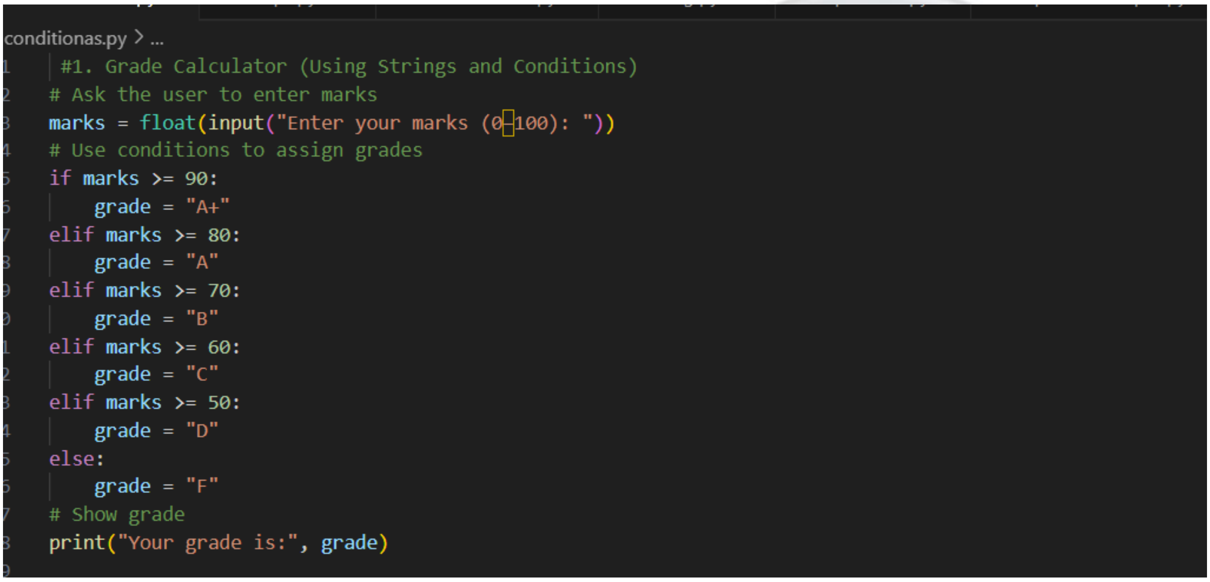


```
Enter a word or sentence: I am doing intership has a cs student
Vowels: 11
Consonants: 19
PS D:\New folder (2)>
```

## Task#07:Week 1 Tasks – Conditionals

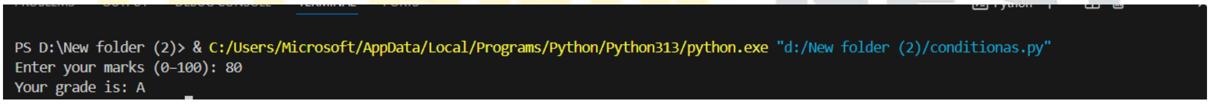
## Q#1: Grade calculator.

## Code:



```python
#1. Grade Calculator (Using Strings and Conditions)
# Ask the user to enter marks
marks = float(input("Enter your marks (0-100): "))
# Use conditions to assign grades
if marks >= 90:
    grade = "A+"
elif marks >= 80:
    grade = "A"
elif marks >= 70:
    grade = "B"
elif marks >= 60:
    grade = "C"
elif marks >= 50:
    grade = "D"
else:
    grade = "F"
# Show grade
print("Your grade is:", grade)
```

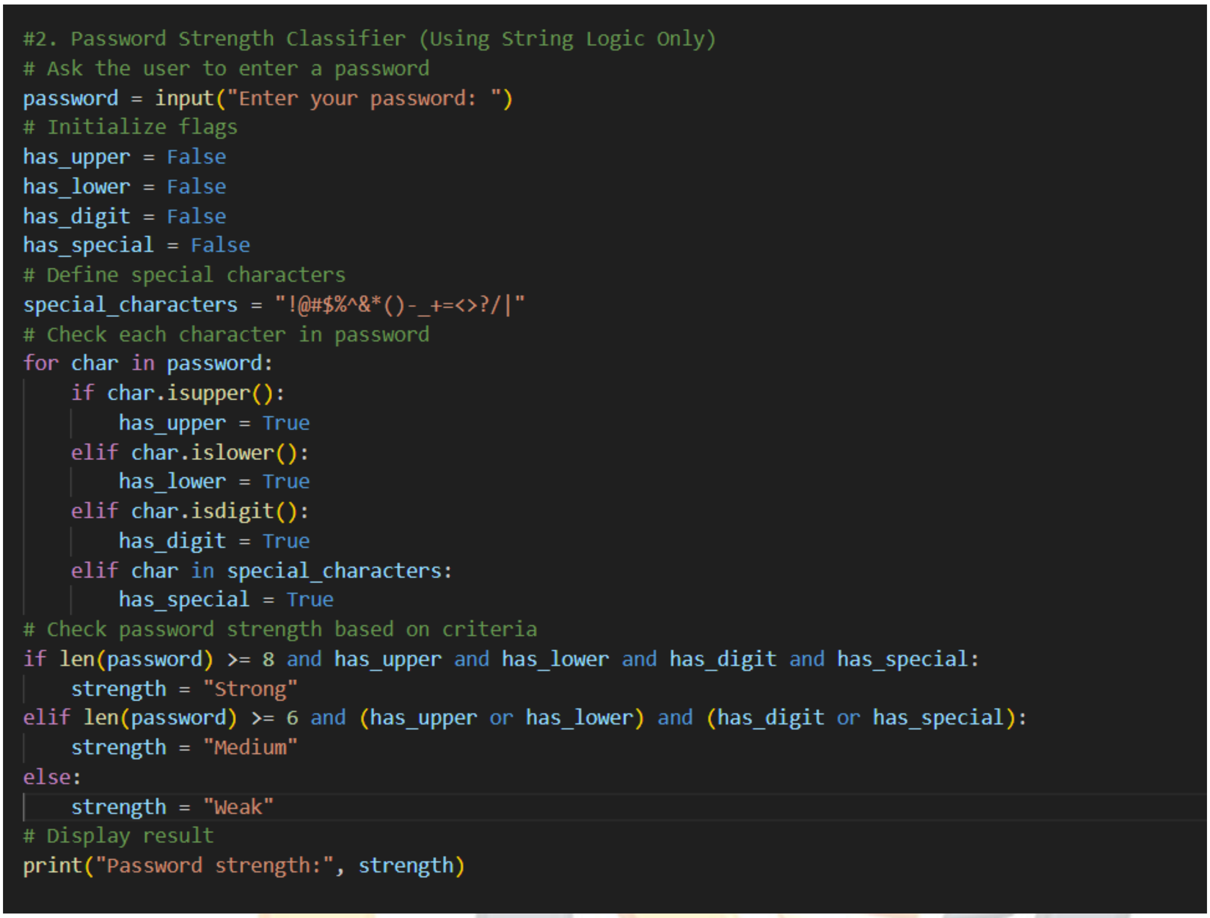
## Solution:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/conditionas.py"
Enter your marks (0-100): 80
Your grade is: A
```

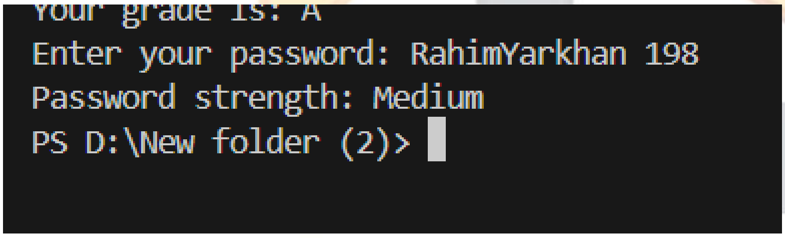
## Q#02: Password strength classifier.

## Code:



```python
#2. Password Strength Classifier (Using String Logic Only)
# Ask the user to enter a password
password = input("Enter your password: ")
# Initialize flags
has_upper = False
has_lower = False
has_digit = False
has_special = False
# Define special characters
special_characters = "!@#$%^&*()-_+=<>?/|"
# Check each character in password
for char in password:
    if char.isupper():
        has_upper = True
    elif char.islower():
        has_lower = True
    elif char.isdigit():
        has_digit = True
    elif char in special_characters:
        has_special = True
# Check password strength based on criteria
if len(password) >= 8 and has_upper and has_lower and has_digit and has_special:
    strength = "Strong"
elif len(password) >= 6 and (has_upper or has_lower) and (has_digit or has_special):
    strength = "Medium"
else:
    strength = "Weak"
# Display result
print("Password strength:", strength)
```

**Output:**

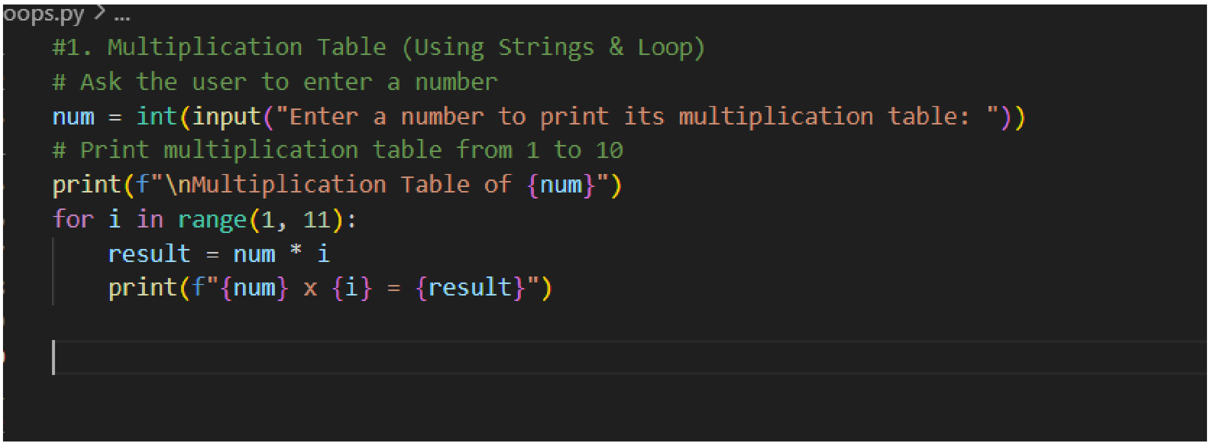


```
Your grade is: A
Enter your password: RahimYarkhan 198
Password strength: Medium
PS D:\New folder (2)>
```

## Task#08:Week 1 Tasks – Loops

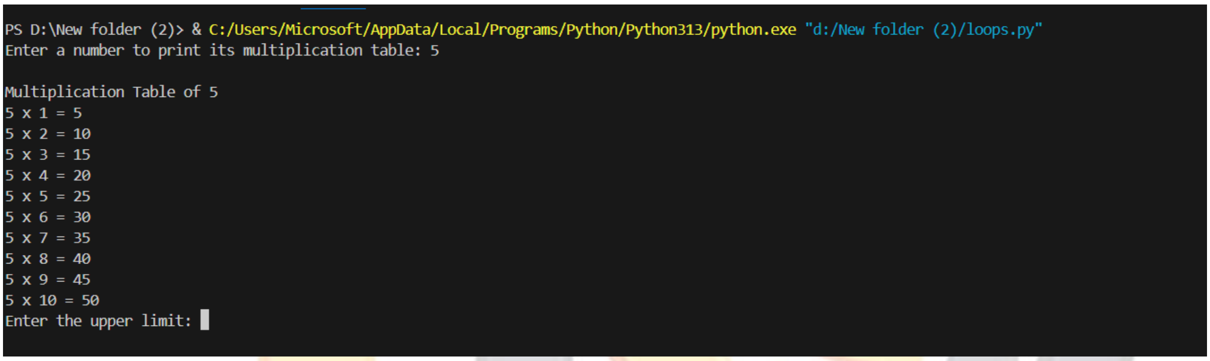
## Q#01: Multiplication table.

## Code:



```
oops.py > ...
#1. Multiplication Table (Using Strings & Loop)
# Ask the user to enter a number
num = int(input("Enter a number to print its multiplication table: "))
# Print multiplication table from 1 to 10
print(f"\nMultiplication Table of {num}")
for i in range(1, 11):
    result = num * i
    print(f"{num} x {i} = {result}")
```

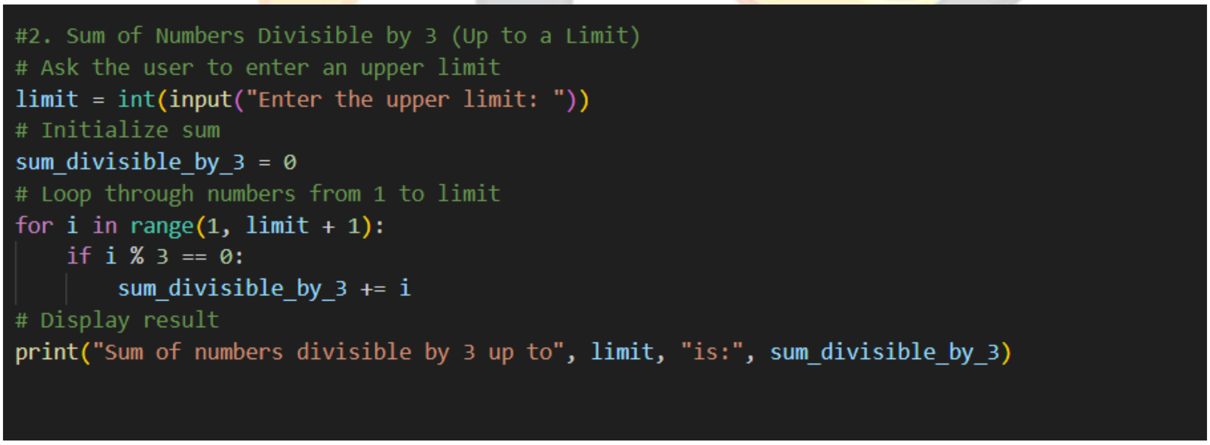
## Solution:



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/loops.py"
Enter a number to print its multiplication table: 5

Multiplication Table of 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
Enter the upper limit: 
```

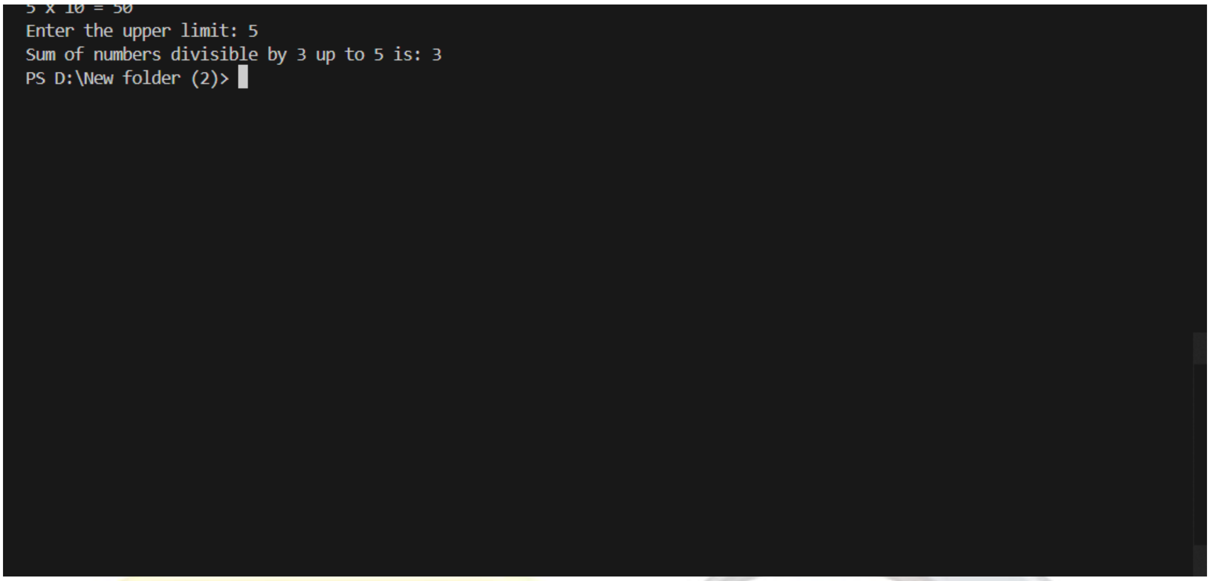
## Q#02: 2. Sum numbers divisible by 3.

## Code:



```
#2. Sum of Numbers Divisible by 3 (Up to a Limit)
# Ask the user to enter an upper limit
limit = int(input("Enter the upper limit: "))
# Initialize sum
sum_divisible_by_3 = 0
# Loop through numbers from 1 to limit
for i in range(1, limit + 1):
    if i % 3 == 0:
        sum_divisible_by_3 += i
# Display result
print("Sum of numbers divisible by 3 up to", limit, "is:", sum_divisible_by_3)
```

## Solution:

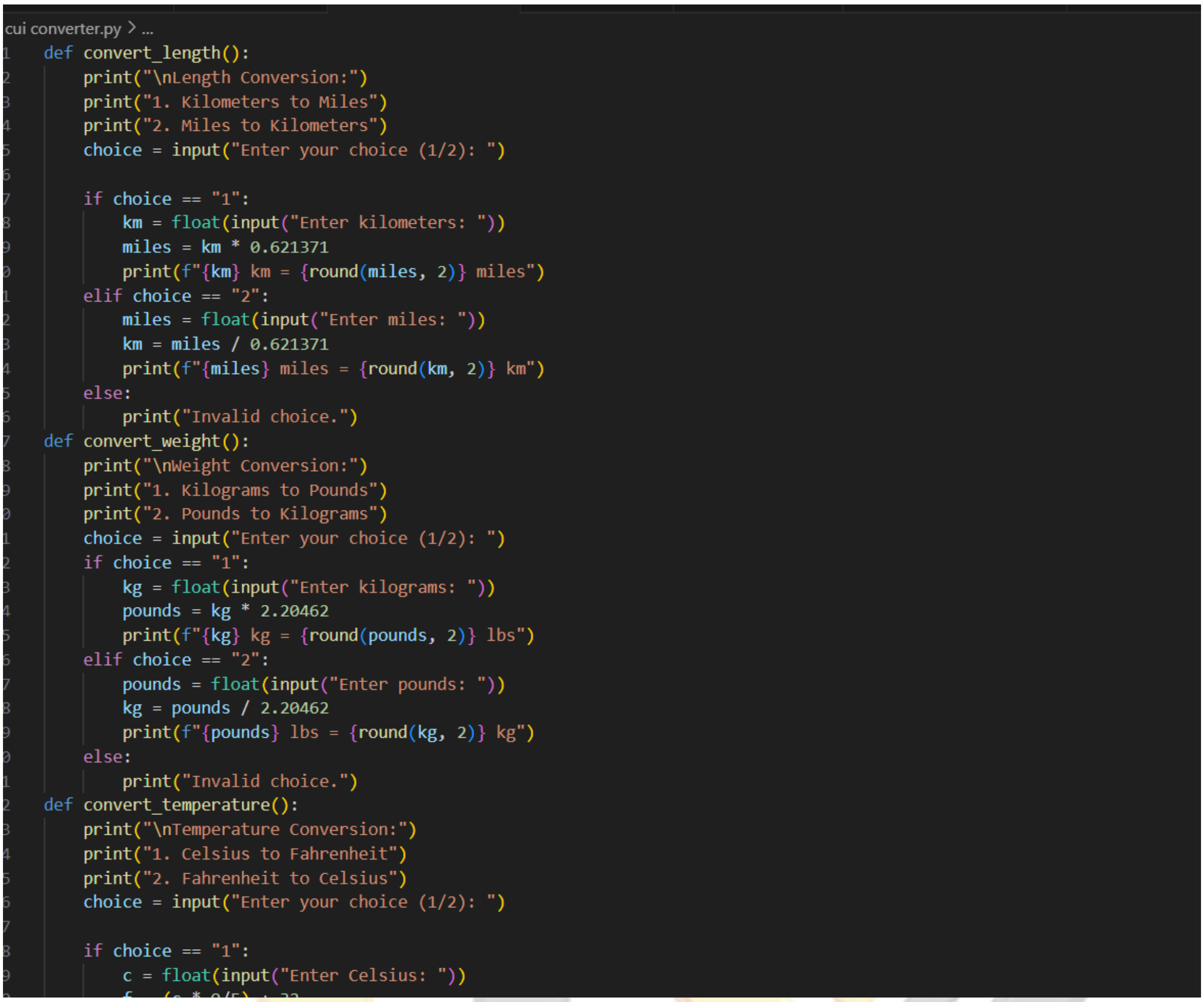


```
5 x 10 = 50
Enter the upper limit: 5
Sum of numbers divisible by 3 up to 5 is: 3
PS D:\New folder (2)> 
```

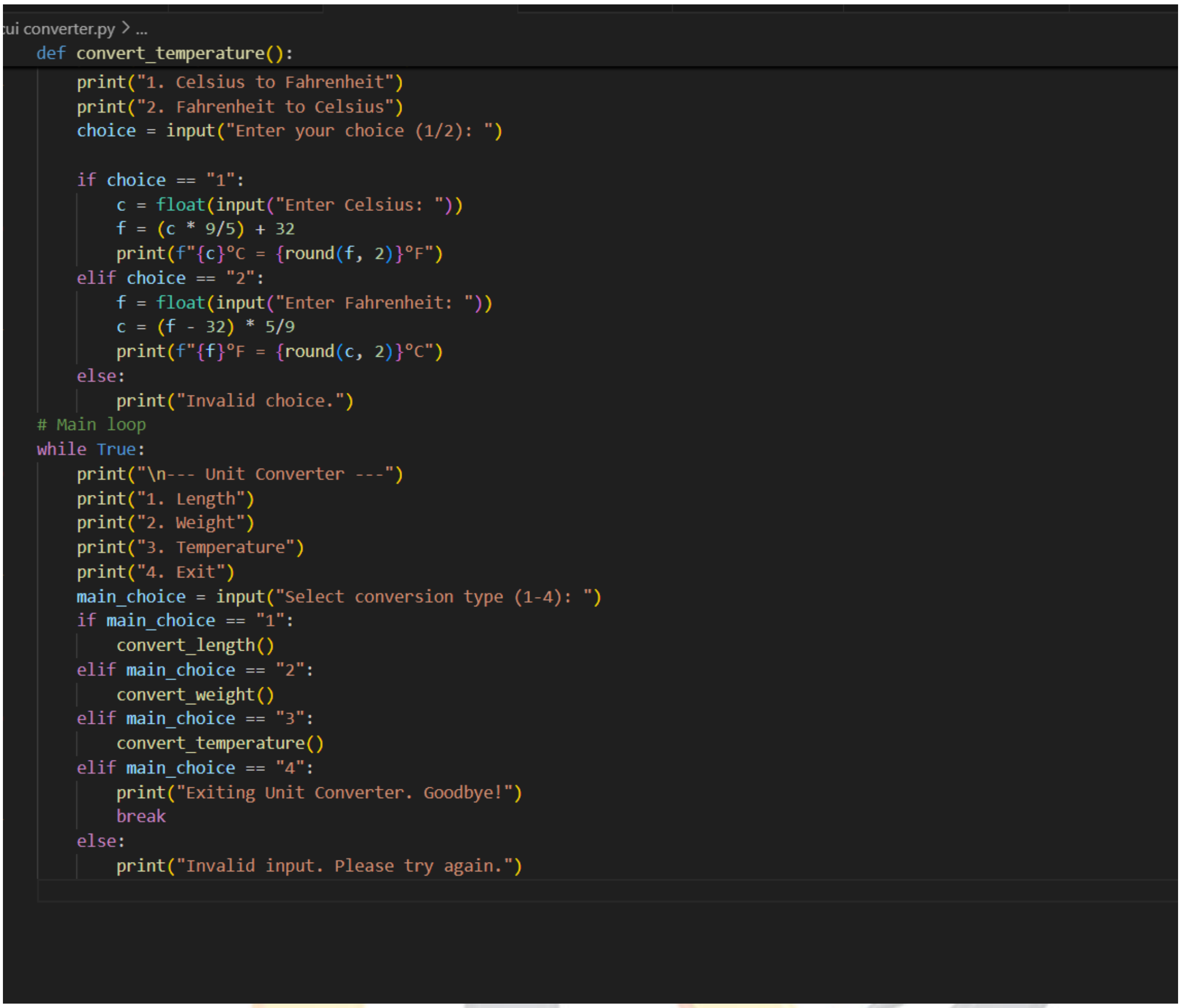
## Task#09:Week 1 Weekly Challenge (Hard)

## Q#01: CLI Unit Converter: length, weight, temperature menus + loops & conditionals.

## Code:

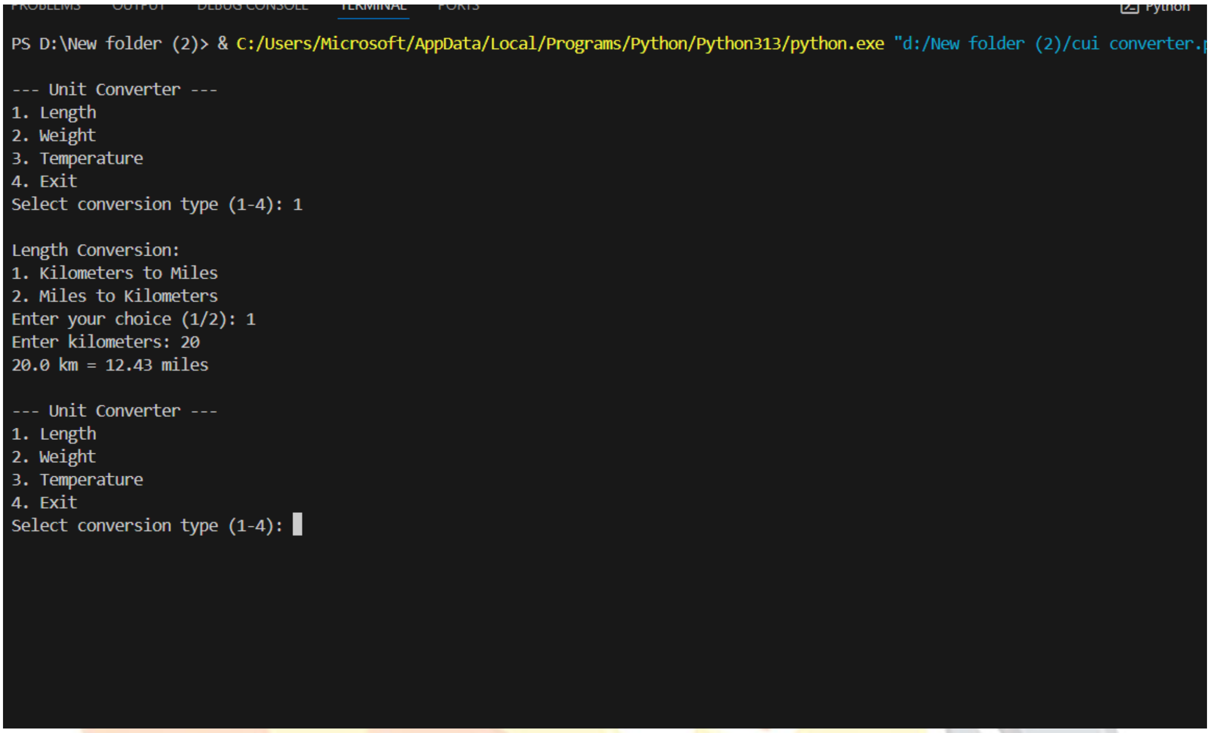


```python
def convert_length():
    print("\nLength Conversion:")
    print("1. Kilometers to Miles")
    print("2. Miles to Kilometers")
    choice = input("Enter your choice (1/2): ")

    if choice == "1":
        km = float(input("Enter kilometers: "))
        miles = km * 0.621371
        print(f"{km} km = {round(miles, 2)} miles")
    elif choice == "2":
        miles = float(input("Enter miles: "))
        km = miles / 0.621371
        print(f"{miles} miles = {round(km, 2)} km")
    else:
        print("Invalid choice.")
def convert_weight():
    print("\nWeight Conversion:")
    print("1. Kilograms to Pounds")
    print("2. Pounds to Kilograms")
    choice = input("Enter your choice (1/2): ")
    if choice == "1":
        kg = float(input("Enter kilograms: "))
        pounds = kg * 2.20462
        print(f"{kg} kg = {round(pounds, 2)} lbs")
    elif choice == "2":
        pounds = float(input("Enter pounds: "))
        kg = pounds / 2.20462
        print(f"{pounds} lbs = {round(kg, 2)} kg")
    else:
        print("Invalid choice.")
def convert_temperature():
    print("\nTemperature Conversion:")
    print("1. Celsius to Fahrenheit")
    print("2. Fahrenheit to Celsius")
    choice = input("Enter your choice (1/2): ")

    if choice == "1":
        c = float(input("Enter Celsius: "))
        f = (c * 9/5) + 32
```



```python
def convert_temperature():
    print("1. Celsius to Fahrenheit")
    print("2. Fahrenheit to Celsius")
    choice = input("Enter your choice (1/2): ")

    if choice == "1":
        c = float(input("Enter Celsius: "))
        f = (c * 9/5) + 32
        print(f"{c}°C = {round(f, 2)}°F")
    elif choice == "2":
        f = float(input("Enter Fahrenheit: "))
        c = (f - 32) * 5/9
        print(f"{f}°F = {round(c, 2)}°C")
    else:
        print("Invalid choice.")
# Main loop
while True:
    print("\n--- Unit Converter ---")
    print("1. Length")
    print("2. Weight")
    print("3. Temperature")
    print("4. Exit")
    main_choice = input("Select conversion type (1-4): ")
    if main_choice == "1":
        convert_length()
    elif main_choice == "2":
        convert_weight()
    elif main_choice == "3":
        convert_temperature()
    elif main_choice == "4":
        print("Exiting Unit Converter. Goodbye!")
        break
    else:
        print("Invalid input. Please try again.")
```

**Output:**



```
PS D:\New folder (2)> & C:/Users/Microsoft/AppData/Local/Programs/Python/Python313/python.exe "d:/New folder (2)/cui converter.

--- Unit Converter ---
1. Length
2. Weight
3. Temperature
4. Exit
Select conversion type (1-4): 1

Length Conversion:
1. Kilometers to Miles
2. Miles to Kilometers
Enter your choice (1/2): 1
Enter kilometers: 20
20.0 km = 12.43 miles

--- Unit Converter ---
1. Length
2. Weight
3. Temperature
4. Exit
Select conversion type (1-4): 
```